Issues Related to Data Sharing

Anthony W. Isenor
Defence R&D Canada – Atlantic
anthony.isenor@drdc-rddc.gc.ca
DRDC Atlantic SL 2005-183

Introduction

The FORCEnet (Fn) vision of data sharing between systems will likely encounter many issues related to both the structure and content of the data. This paper briefly outlines some key issues that Fn should expect to encounter.

Conceptually, when we consider data sharing between two independent systems or nodes, the data communication between the nodes requires two data transformations. For example, node 1 sends data to node 2, with node 2 responsible for transforming the data into a usable structure for the node (first transformation). Similarly, when node 2 sends data to node 1 a second transformation is required. In a network of n nodes, a total of n(n-1) transformations are required to account for communication among all nodes.

Incorporating a central structure (see Figure 1) for data transfer reduces the number of transformations. In this model, all nodes convert to and from the central structure, reducing the transformation requirement to 2n.

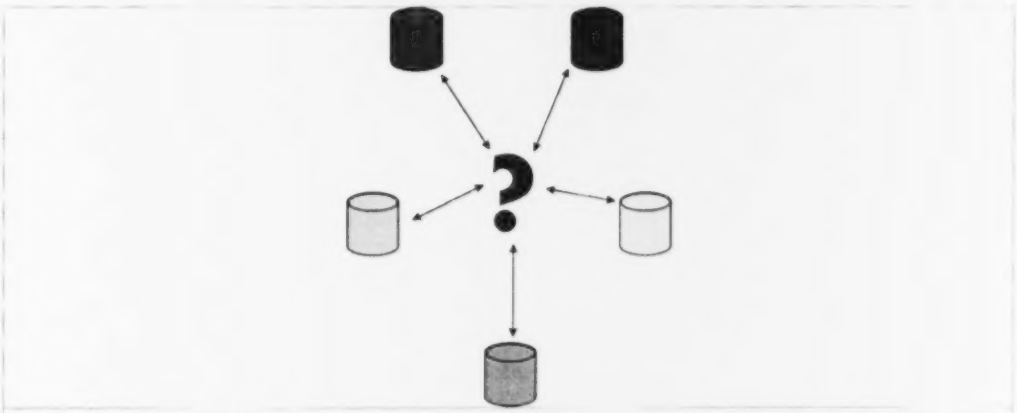

Figure 1. The data sharing (represented by arrows) between nodes (represented by cylinders) in a network can be accomplished using a central structure or a central database, as depicted by the question mark. This data-sharing model requires 2n transformations (where n is the number of nodes) to ensure communication between all nodes.

Issues Related to Data Transformation

The communication between nodes includes a component related to the transformation of the data to meet the requirements of the receiving system (Figure 2). Constructing a transformation application requires governance for both systems involved in the transformation. The transformation process also introduces several issues related to the actual manipulation of the data structure and data content. Four issues are of particular importance and may be summarized as: codes, units, metadata and data alteration. The relative importance of the individual issues will depend on the particular situation.

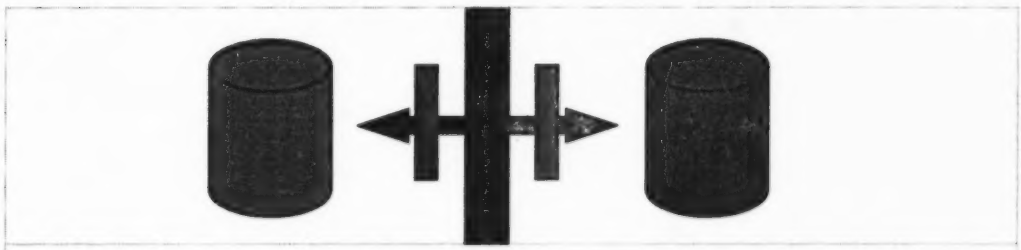

Figure 2. The illustration shows two nodes or systems (cylinders) sharing data via the arrow. Each system has an individual client (shown by vertical bars of similar colour to the nodes). The larger blue vertical bar represents the transformation process.

Codes

A code is a form of condensed information that is used in many data systems. As examples, a code could apply to environmental measurements like temperature (e.g., the code could be TEMP), ships (e.g., the code might be Fhfx) or instruments (e.g., hms). When data transformations take place, the systems must ensure that a method exists for translating between the code languages (sometimes referred to as a vocabulary) used in the two systems. Definitions are also typically associated with the code and may be important for knowing characteristics of the data. Such definitions would indicate specifics of the data that are important for proper interpretation and use of the data.

Units

Units will pose problems for any system that utilizes physical measurements of quantities. Each independent node will likely describe units in different ways (e.g., Celsius, C, deg C, degrees C). In some nodes, the unit may not be explicitly stored with the data, but rather assumed based on the local applications that produce the data. When transformations take place, conversion factors will be an issue. Consistent conversions will be required for all nodes. Otherwise, data values may be subtly changed from their intended value. In some cases, conversions utilize other measured parameters. In these cases, the inverse conversion can only be applied if the other parameter is also available.

Metadata

Metadata assists the user in utilizing a data set. The codes and units that accompany the data are a form of metadata. In particular, codes are a type of usage vocabulary. Particular subject areas often develop usage vocabularies for the subject. Such vocabularies could involve platforms such as ships or aircraft, instrumentation, sensors, data, etc. However, usage vocabularies are only part of the metadata that is important in the data transformation process. Discovery vocabularies are also important in any network that involves an active discovery of data, as opposed to nodes simply sharing all available data. A discovery vocabulary provides categorization of the usage vocabulary, in a language familiar to the subject community. The metadata structure will also be important, as the metadata will only be understood if the structure is decipherable by the receiving node. There are recognized international metadata structures that may be utilized, such as the ISO 19115.

Data Alteration

Data alteration refers to the process of changing the data in some way to meet the requirements of the receiving system. As examples, data may be added (or omitted) to conform to the design of a receiving database, values may be rounded as a result of storage in a system, or conversions may be applied (although data corrections also represent a data alteration, it is unlikely the correction would be applied in the transformation). Such alterations are necessary for the receiving system, but do result in a slightly different representation of the data. This can potentially lead to data incest. Data tagging may be employed in an attempt to deal with potential incest problems.

XML Example

Figure 3 shows a section of an XML document that contains data streaming into a LC2IEDM[1] table. The example corresponds to a single data record being loaded into the *vertical_distance* table using the XML structure used in the Operational Context Exchange Service (OCXS[2]) (developed by the US Naval Undersea Warfare Center, NUWC). Table 1 provides examples of the issues mentioned above, by using the data from Figure 3.

Conclusion

In the networked environment, it is unlikely that data availability will be a problem. A more likely issue will involve understanding the data, both in terms of its structure (i.e., the syntax) and its meaning (i.e., the semantics). Any method that deals with the transfer of data between disparate nodes will have to deal with both the syntax and semantics surrounding data transfer.

---

[1] LC2IEDM - Land Command and Control Information Exchange Data Model

[2] Burkley, Frederick G. Integrated Computing Environment, Operational Context Exchange Service. October 30, 2002.

```
<VerticalDistanceTable>
 <VerticalDistance>
  <VerticalDistanceId>-2147483647</VerticalDistanceId>
  <VerticalDistanceCategoryCode>LOCSUR</VerticalDistanceCategoryCode>
  <VerticalDistanceDimension>256.6</VerticalDistanceDimension>
  <VerticalDistancePrecisionCode>10M</VerticalDistancePrecisionCode>
  <OwnerId>1</OwnerId>
  <UpdateSeqnr>1</UpdateSeqnr>
 </VerticalDistance>
</VerticalDistanceTable>
```

Figure 3. An XML snippet that shows the OCXS structure being loaded into a single record in a LC2IEDM table. In this example, the two systems shown in Figure 2 are represented by OCXS and LC2IEDM.

Table 1. This table provides some examples that relate the transformation issue to the XML content in Figure 3. Realize that all of these issues are exposed by the insertion of a single record into a single table in the LC2IEDM database.

ISSUE	VALUE FROM FIGURE 3	RELATING THE ISSUE TO THE FIGURE 3 VALUE
Code	LOCSUR	This code means something specific to the originating system. The receiving node must understand such specifics.
Units	256.6	This value has an assumed unit. Any data transfer must ensure that the receiving node properly understands the unit.
Metadata	(consider the structure of the XML rather than a particular value)	The definitions for the structure must be available. This would include the definition of tag names, tag order, definitions of intended content, etc.
Data Alteration - Tagging Example	<VerticalDistanceId>-2147483647</VerticalDistanceId>	The content is a unique identifier for the record. The procedure for generating the identifier must be either consistent across all nodes, or known and understood by the nodes.
Data Alteration - Tagging Example	<OwnerId>1</OwnerId> <UpdateSeqnr>1</UpdateSeqnr>	This tagging identifies the data owner and an update sequence number (the update sequence number may be thought of as a version number).
Governance	<VerticalDistanceCategoryCode>	This is just one of the tags in the XML snippet. The tag represents a field in an LC2IEDM table. LC2IEDM governance documentation would address issues such as why this particular naming convention was adopted, any previous versions of the field name, example usage, what is the reasoning behind including this category code, etc.